

PEMILIHAN *CROSSOVER* PADA ALGORITMA GENETIKA UNTUK PROGRAM APLIKASI PENGENALAN KARAKTER TULISAN TANGAN

Judah Suryaputra ¹⁾ Chairisni Lubis ²⁾ Tri Sutrisno ³⁾

Teknik Informatika, FTI Universitas Tarumanagara
Jalan Letjen S Parman 1, Jakarta Barat 11440, Indonesia
email: judahsuryaputra@gmail.com

ABSTRACT

Handwriting recognition system using genetic algorithm is an Optical Character Recognition system which receives input in the form of handwritten image in scanned box and produces output in the form of characters from the handwriting. Writing can be uppercase, lowercase, or numbers. The designed system consists of five main processes: preprocessing the input image, vertical and horizontal line segmentation, line segmentation and character with automatic cropping, resizing template, and character recognition using genetic algorithm. Preprocessing of input image consists of grayscale process, and thresholding. Genetic algorithm is used to find characters from the character image obtained by comparing the image with the chromosome of the train data. To use the genetic algorithm method, given the process of resizing the template first so that the image size of the characters with the same template. This system has a success rate of character segmentation of 100% and success rate on character recognition with genetic algorithm of 89,027% with one point crossover, 90,43% with two point crossover, 90,72% with uniform crossover.

Keywords

Automatic Cropping, Optical Character Recognition, Character Segmentation, Genetic Algorithm, Handwriting Recognition.

1. Pendahuluan

Dari tahun ke tahun, perkembangan di bidang teknologi semakin pesat dan teknologi ke-pintaran buatan sudah banyak dikenal dan digunakan. Ke-pintaran buatan adalah suatu sistem yang dapat membantu manusia dalam menyelesaikan suatu masalah. Suatu ke-pintaran buatan dapat menyelesaikan pekerjaan jika memiliki

program yang telah didefinisikan terlebih dahulu. Dengan adanya program yang telah ada, ke-pintaran buatan dapat melakukan berbagai tugas tergantung dari program yang telah diberikan, seperti mengenali karakter tulisan tangan manusia.

Untuk mendapatkan karakter, dibutuhkan cropping dengan menggunakan metode histogram automatic cropping. Metode ini terdiri dari beberapa proses. Dimulai dari menghitung jumlah piksel hitam dalam tiap baris secara horisontal untuk memisahkan per baris dan menghitung jumlah piksel hitam dalam satu kolom untuk memisahkan per karakter. Sehingga didapatkan nilai array 1 dimensi yang disebut sebagai kromosom.

2. Dasar Teori

Beberapa teori dan metode yang digunakan dalam perancangan sistem adalah *Pre-processing*, *Horizontal Automatic Cropping*, dan *Genetic Algorithm*.

2.1 Preprocessing

Preprocessing adalah salah satu tahap awal yang harus dilakukan sebelum citra dapat diproses lebih lanjut. Data yang diinput harus diolah terlebih dahulu guna menghilangkan bagian-bagian yang tidak dibutuhkan dalam proses pengenalan. Proses *preprocessing* yang dilakukan adalah *grayscale* dan *thresholding*.

2.2 Automatic Cropping

Automatic Cropping digunakan untuk memotong citra yang telah dihilangkan kotaknya menjadi citra karakter. Langkah-langkah *histogram automatic cropping* adalah:

1. Segmentasi baris

Menentukan setiap baris kalimat dalam teks tulisan tangan. Input citra yang digunakan merupakan teks tulisan tangan yang dimasukkan ke dalam sistem dengan media input scanner. Piksel ditelusuri dari pojok kiri atas sampai kanan bawah. Jika baris piksel pertama yang ditelusuri kosong atau tidak memiliki informasi mengenai objek yang akan dikenali. Jika baris piksel memiliki titik hitam maka baris yang memiliki titik hitam tersebut akan dihitung jumlah piksel hitam secara horizontal pada setiap baris piksel. Persamaan untuk mendapatkan piksel hitam secara horizontal adalah sebagai berikut:

$$Ph[i] = \sum_{j=1}^M S[i, j]$$

Keterangan:

S = ukuran matriks citra biner

M = banyaknya baris citra

Ph[i] = proyeksi horizontal yang berwarna hitam

i = baris

j = kolom

2. Segmentasi karakter

Segmentasi karakter adalah cara menghitung karakter agar dapat dipotong, sedikit sama dengan cara mencari nilai per baris kalimat (horizontal histogram). Untuk mendapatkan baris kalimat digunakan horizontal histogram sedangkan untuk mencari per huruf atau karakter dapat digunakan vertikal histogram. Vertikal histogram dapat dihitung pada setiap kolom kalimat. Setelah semua nilai pada setiap kolom baris kalimat sudah didapatkan, maka dapat dibuat vertikal histogram. Perhitungan untuk menghitung titik hitam pada setiap kolom dapat dijelaskan sebagai berikut:

$$Pv[j] = \sum_{i=1}^N S[i, j]$$

Keterangan:

S = ukuran matriks citra biner

N = banyaknya kolom citra

Pv[j] = proyeksi vertikal yang berwarna hitam

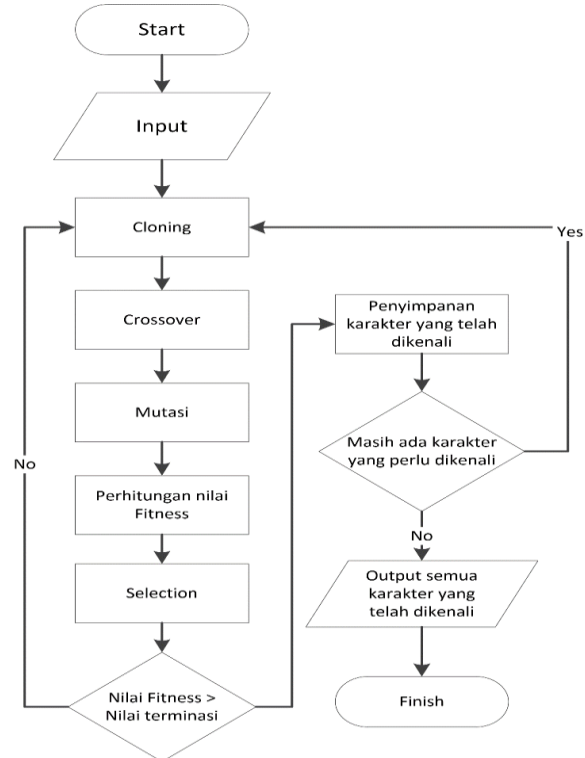
i = baris

j = kolom

2.3 Genetic Algorithm

Algoritma genetika adalah cabang dari algoritma evolusi merupakan metode adaptif yang biasa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah seleksi. Algoritma ini akan bekerja dalam sebuah populasi yang terdiri dari individu-individu (kromosom) yang masing-masing individu merepresentasikan sebuah solusi yang mungkin bagi

persoalan yang ada. Kromosom dalam algoritma genetika merupakan suatu vektor yang berisi nilai biner hasil dari binerisasi citra. Kromosom terdiri dari 2 yaitu parent dan child. Parent merupakan kromosom data uji yang akan dilakukan cloning sedangkan child merupakan hasil dari kromosom data uji yang telah dilakukan proses algoritma genetika yaitu cloning dan mutasi.



Gambar 1 Flowchart Genetic Algorithm

2.3.1 Cloning

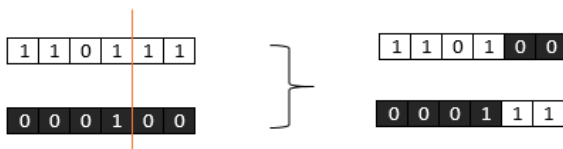
Tahap paling pertama dalam algoritma genetika adalah cloning. Cloning merupakan suatu proses dimana kromosom yang sudah diekstrak dari citra biner, diduplikasi supaya dapat diproses lebih lanjut. Kromosom hasil cloning dilakukan mutasi untuk setiap hasil cloning yang bertujuan untuk membangkitkan kromosom baru yang berbeda dengan cara mengganti nilai gen bernilai 0 menjadi 1 dan sebaliknya.

2.3.2 Crossover

Pindah silang dilakukan dengan harapan kromosom-kromosom baru akan mempunyai bagian yang baik dari kromosom-kromosom lama dan tidak menutup kemungkinan menjadi kromosom-kromosom yang lebih baik. Jenis-jenis operator crossover :

- a. One Point Crossover

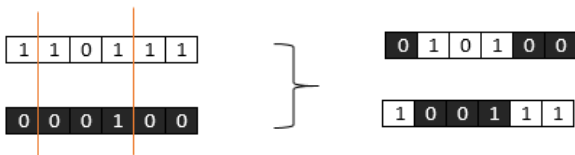
Crossover ini melakukan penukaran gen-gen dari satu kromosom dengan kromosom lain untuk menghasilkan kromosom baru melalui satu titik potong. Titik potong didapatkan dengan cara membangkitkan random number dengan batasan 1 sampai n (panjang kromosom). Random number yang dihasilkan akan dijadikan sebagai titik potong kromosom. Sebagai contoh dua buah kromosom memiliki panjang 6 dan random number yang dihasilkan adalah 4, maka gen 1 sampai 4 akan dipotong dengan gen 5 sampai 6. Kemudian gen 1 sampai 4 pada kromosom pertama akan dilakukan crossover dengan gen 5 dan 6 pada kromosom kedua, dan sebaliknya. Ilustrasi persilangan one point crossover dapat dilihat pada Gambar 5.



Gambar 2 One Point Crossover

b. Two Point Crossover

Crossover ini melakukan penukaran gen dari satu kromosom dengan kromosom lain untuk menghasilkan kromosom baru melalui beberapa titik potong. Sama dengan one point crossover, perbedaannya adalah two point crossover membangkitkan 2 random number sebagai titik potong kromosom yang berarti satu kromosom di potong menjadi 3 bagian yang kemudian disilangkan dengan kromosom lawannya.[3]



Gambar 3 Two Point Crossover

c. Uniform Crossover

Crossover ini melakukan penukaran gen-gen dari satu kromosom dengan kromosom lain melalui tiap index berdasarkan probabilitas. Setiap gen memiliki probabilitas seperti koin. Jika yang muncul kepala, maka gen akan ditukar dan jika muncul buntut, maka posisi gen tetap. Uniform crossover dapat menghasilkan rekombinasi kromosom yang lebih baik. Secara keseluruhan, uniform crossover tidak terganggu dengan kromosom yang panjang. Hal ini memungkinkan uniform crossover untuk tampil efektif. Uniform crossover memiliki tambahan properti yang memiliki daya eksplorasi lebih banyak daripada multi point

crossover. Untuk membuktikannya, ambil contoh kasus ekstrem di mana salah satu kromosom yang bernilai 0 semua dan kromosom lain bernilai 1 semua. Uniform crossover dengan jelas bisa menghasilkan keturunan yang memiliki nilai 0 dan 1 secara acak sedangkan crossover 1 titik dan crossover 2 titik lebih terbatas.[4]



Gambar 4 Uniform Crossover

2.3.3 Mutasi

Proses mutasi dilakukan dengan cara mengganti satu gen dalam kromosom yang terpilih secara acak dengan suatu nilai baru yang didapat secara acak.

2.3.4 Fitness Function

Nilai fitness dalam sebuah algoritma genetik menggambarkan tingkat konvergensi keoptimalan algoritma dimana yang diharapkan adalah nilai fitness yang optimal dalam hal ini angka tertinggi ialah nilai terbaik. Dalam evolusi dunia nyata, individu bernilai fitness tinggi akan bertahan hidup, sedangkan yang memiliki nilai fitness rendah akan gugur atau mati. Pada algoritma genetik, fitness biasanya dapat berupa fungsi objektif dari masalah yang akan dioptimalisasi. Kromosom-kromosom diseleksi menurut nilai fitness masing-masing Kromosom yang kuat mempunyai kemungkinan tinggi untuk bertahan hidup pada generasi berikutnya. Perhitungan fitness function dapat dilakukan sebagai berikut:[5]

$$F.F = \frac{1}{\sum_{i=1}^n (\alpha_i - \beta_i)^2 + 1}$$

Keterangan:

- α = kromosom data latih
- β = kromosom data uji
- n = banyak gen dalam satu kromosom

i. Selection

Seleksi ada proses pemilihan kromosom yang memiliki nilai fitness tertinggi dibandingkan dengan kromosom lainnya. Kromosom yang terpilih dijadikan sebagai parent yang kemudian akan diulangi proses mulai dari cloning sampai selection kembali sampai nilai fitness dari kromosom yang didapatkan melebihi batas threshold.

3. Rancangan Penelitian

Penelitian akan dilakukan dengan menyimpan citra latih sebanyak 13 untuk tiap karakter A-Z, a-z, dan 0-9. Tulisan tangan yang ditulis oleh Judah Suryaputra. Pengambilan data dilakukan dengan cara memindai kertas yang berisi tulisan tangan data latih seperti gambar 5



Gambar 5 Contoh Data Input

Penelitian dilakukan dengan mengambil data uji sebanyak 50 citra uji dengan setiap data dilakukan pengujian sebanyak lima kali untuk setiap jenis *crossover*. Sehingga total percobaan dilakukan sebanyak 750 kali.

4. Pengujian

Pengujian dilakukan dengan sepuluh citra input yang berbeda. Pengujian dilakukan pada kertas berwarna putih yang telah diberikan kotak berwarna hitam sebanyak 5x7. Kotak yang diberikan berukuran 2cm X 2cm. Hasil dari pengujian yang dilakukan, pada segmentasi karakter didapatkan persentase keberhasilan sebesar 100% sedangkan pada pengenalan karakter dengan *genetic algorithm* didapatkan persentase keberhasilan sebesar 88,92% dengan *one point crossover*, 90,43% dengan *two point crossover*, 90,72% *uniform crossover*.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Kesimpulan yang diperoleh dari hasil pengujian "Pemilihan Crossover pada Algoritma Genetika untuk Program Aplikasi Pengenalan Tulisan Tangan" yang telah dilakukan adalah sebagai berikut:

1. Segmentasi karakter tulisan tangan dengan batas kotak sebesar 2 cm X 2 cm menghasilkan persentase keberhasilan sebesar 100%.
2. Apabila terdapat noda di dalam area kotak penulisan, hasil pengenalan akan menurun.
3. Kualitas dari hasil scan sangat menentukan hasil dari pengenalan karakter. Bila kualitas dari hasil scan kurang baik, atau miring, hasil proses pengenalan akan menurun.

4. Dari ketiga cara *crossover*, *crossover* dengan menggunakan *uniform crossover* memiliki tingkat kekonsistenian tertinggi.

5. Ketiga pilihan cara *crossover* memiliki tingkat keakuratan yang tidak jauh berbeda yaitu 89,027% untuk *one point crossover*, 90,43% untuk *two point crossover*, dan 90,728% untuk *uniform crossover* sehingga dapat disimpulkan *uniform crossover* memiliki hasil terbaik.

5.2 Saran

Saran yang diberikan untuk perancangan program ini dan juga penelitian berikutnya adalah:

1. Menambahkan fitur untuk meluruskan citra input apabila didapati citra input yang miring.
2. Menambahkan metode untuk meningkatkan kualitas citra agar hasil pengenalan lebih baik.
3. Menambahkan metode pengenalan yang dapat mengenali karakter yang apabila ditulis secara miring.

REFERENSI

- [1] Zaidi Razak dan Khansa Zulkiflee, A Real Time Line Segmentation Algorithm For an Offline Overlapped Handwritten Jawi haracter Recognition Chip, (Kuala Lumpur: Faculty of Computer Science and Information Technology University of Malaya), hlm. 173.
- [2] Poli, Riccardo, and William B. Langdon. "Genetic programming with one-point crossover." *Soft Computing in Engineering Design and Manufacturing*. Springer, London, 1998. 180-189.
- [3] Kaya, Yilmaz, and Murat Uyar. "A novel crossover operator for genetic algorithms: Ring crossover." *arXiv preprint arXiv:1105.0355* (2011).
- [4] Hu, Xiao-Bing, and Ezequiel Di Paolo. "An efficient genetic algorithm with uniform crossover for air traffic control." *Computers & Operations Research* 36.1 (2009): 245-259.
- [5] A. De Jong, Kenneth. A Formal Analysis of the Role of Multi-Point Crossover in Genetic Algorithms, Fairfax: *Annals of Mathematics and Artificial Intelligence*, 1992.

Judah Suryaputra, mahasiswa S1, program studi Teknik Informatika, Fakultas Teknologi Informasi Universitas Tarumanagara.

Dra. Chairisni Lubis M.Kom, memperoleh gelar Dra. Dari Matematika dan Ilmu Pengetahuan Alam Universitas Indonesia dan gelar M.Kom dari Universitas Indonesia . Saat ini sebagai Dosen program studi Teknik Informatika, Universitas Tarumangara.

Tri Sutrisno S.Si.,M.Sc. memperoleh S.Si dari Universitas Diponegoro kemudian memperoleh gelar M.Sc. dari Universitas Gadjah Mada. Saat ini sebagai Dosen program studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Tarumanagara.